

REMARKS

Applicant's representative expresses appreciation for the in person interview conducted on September 16, 2009. The amendments made by this response and the remarks are consistent with that which was discussed during the interview. During the interview, the examiner agreed that the cited art fails to teach or suggest each limitation of the independent claims.

The Non-Final Office Action mailed July 21, 2009 considered claims 1, 3-6, 8-16, 18-20, and 22-35 each of which were rejected under 35 U.S.C. § 103(c) as being unpatentable over Cynerman and Jerger (U.S. Pat. No.: 6,321,334).¹

By this response, claims 20 and 22-31 have been amended so that they are directed to a computer storage medium which does not encompass signals. Claims 22-31 had been erroneously directed to methods.

Traversal of the Prior Art Rejections

The present invention is directed to limiting what an entity (e.g. project files) has access to while it is being built. In other words, the present invention defines a trust level under which the build process will be executed. The specification describes a build process and how the build process's permissions are affected by the entities involved in the build. A build process as described in the specification, therefore, refers to the process of building a software entity (e.g. an application, executable, etc.). As such, a build process could encompass compiling code, linking libraries, etc. Therefore, the claims are directed to defining a permission level under which the build process will perform these actions.

For example, as claimed in claim 1, a system including a build process processor processes one or more build entities. This processing should be construed to encompass compiling, linking, and the like. Before the build process is processed, a policy component is accessed to determine what trust level the build process will be processed with. Each build entity that is being processed as part of the build process has an associated trust level assigned to it. The trust level of the build entity with the lowest trust level is used as the trust level for the build process. Each of the independent claims contains similar limitations as claim 1.

¹ Although the prior art status of the cited art is not being challenged at this time, Applicant reserves the right to challenge the prior art status of the cited art at any appropriate time, should it arise. Accordingly, any arguments and amendments made herein should not be construed as acquiescing to any prior art status of the cited art.

The Cymerman reference describes a scripting tool (Ant) that automates the build process. It is emphasized that Ant is not a build process. Ant simply provides a way to automate the build process that is performed by calling executables using a command prompt. For example, under the heading "What is Ant?" on page 1, Cymerman describes Ant as "a platform-independent scripting tool that lets you construct your build scripts in much the same fashion as the 'make' tool in C or C++." As a scripting tool, Ant runs specified commands using a command prompt. The commands are specified using the target element. The example on page 3 under the title "Simple build process with Ant (simple.xml)" illustrates various targets including init, clean, prepare, compile, and deploy. Ant simply allows a programmer to specify what commands should be run with what parameters using a single XML file. Ant executes these commands as specified. However, the actual building occurs by invoking other executables (e.g. the same functionality can be performed by manually entering the commands in the command prompt. Ant simply stores these commands in an XML file providing for consistency across builds). For example, in order to use Ant, the Java Development Kit (JDK) must also be installed on the system. This is because the executables within the JDK actually perform the build. *See* Pg. 2, "What do I need to use Ant?". Because Ant is not a build process, it cannot teach or suggest any of the limitations of the independent claims. Further, Cymerman makes no mention of accessing a policy to apply a permission level to the build process.

Jerger also does not describe accessing a policy to apply a permission level to the build process. Jerger discloses embodiments for setting permissions for web content based on where the content is retrieved. This is similar to the features provided in current versions of Internet Explorer under the Security tab in the Internet Options window. Specifying permission levels for web content is not similar to specifying permissions for a build process. In particular, the claims specifically state that the level of trust is determined based on the lowest level of trust assigned to a build entity involved in the build process. There is nothing similar to this in either reference.

With specific reference to the limitations of claim 1, neither reference discloses "a policy component that is processed by the build process processor to determine one or more levels of trust within which the build process operates." As addressed above, Ant is not a build process. The only build process that could be implied in either reference would be the executables that are listed within the target element in Ant. However, Cymerman discloses nothing about these build

processes other than that they are called by Ant. Nothing in Ant specifies a level of trust to be applied to the build process. Further, Applicant submits that the examiner's combination of Cymerman with Jerger is unreasonable primarily because Cymerman only tangentially addresses build processes and Jerger has nothing to do with build processes. Finally, the examiner has failed to provide a valid reason for combining the teachings of the two references. The stated reason is that "one would have been motivated to do so to secure the build process by automatically administering the decision to grant or deny permissions to specific build entities...." This motivation is improper because Ant is not a build process within a reasonable interpretation of the claims. Therefore, modifying Ant with the teachings of Jerger still would not secure the build process since Ant only invokes the executables that perform the build process rather than playing a role during the build process.

Claim 1 also recites that "at build time, a principal permission level under which the build process executes is determined by analyzing the levels of trust associated with each of the build entities, and lowest level of trust of all involved build entities dictates the principal permission level for execution of the build process." Neither reference discloses or suggests this limitation. Jerger does not analyze the levels of trust associated with each of the build entities. In Jerger, permissions are set for an individual file. For example, when a file downloaded from a website wants to run, a permission level is assigned to the file based on what website it is from. There is no determination of which entity has the lowest level of trust because there is only a single entity – the file. Further, in the present invention, the permission level is assigned to the build process based on the level of trust of each build entity. The only possible analogy that can be made in Jerger is to equate the website with the build entity and the file with the build process. Applicant submits, however, that this is a very unreasonable analogy. The build process processes the build entities to create some type of software entity (e.g. an executable). In contrast, the file (which is equated with the build process in this analogy) is itself executed. The file cannot perform any processing on the website. Therefore, this analogy fails. As such, Jerger cannot teach or suggest these limitations. Further, because Cymerman does not address a build process, it cannot be combined with Jerger to show obviousness.

In summary, Applicant submits that the combination of Cymerman and Jerger fails to teach or suggest each limitation of the independent claims. Therefore, the claims are novel and non-obvious in view of the current art.

In view of the foregoing, Applicant respectfully submits that the other rejections to the claims are now moot and do not, therefore, need to be addressed individually at this time. It will be appreciated, however, that this should not be construed as Applicant acquiescing to any of the purported teachings or assertions made in the last action regarding the cited art or the pending application, including any official notice. Instead, Applicant reserves the right to challenge any of the purported teachings or assertions made in the last action at any appropriate time in the future, should the need arise. Furthermore, to the extent that the Examiner has relied on any Official Notice, explicitly or implicitly, Applicant specifically requests that the Examiner provide references supporting the teachings officially noticed, as well as the required motivation or suggestion to combine the relied upon notice with the other art of record.

In the event that the Examiner finds remaining impediment to a prompt allowance of this application that may be clarified through a telephone interview, the Examiner is requested to contact the undersigned attorney at (801) 533-9800.

Dated this 29th day of October, 2009.

Respectfully submitted,

/Brian D. Tucker/

RICK D. NYDEGGER
Registration No. 28,651
BRIAN D. TUCKER
Registration No. 61,550
Attorneys for Applicant
Customer No. 47973